PCT/NO 98/00298
09 530994

REC'D 28 OCT 1998
WIPO PCT

KONGERIKET NORGE
The Kingdom of Norway

Bekreftelse på patentsøknad nr
Certification of patent application no

PRIORITY DOCUMENT
SUBMITTED OR TRANSMITTED IN COMPLIANCE WITH RULE 17.1(a) OR (b)

19975152

► Det bekreftes herved at vedheftede dokument er nøyaktig utskrift/kopi av ovennevnte søknad, som opprinnelig inngitt 1997.11.10

► *It is hereby certified that the annexed document is a true copy of the above-mentioned application, as originally filed on 1997.11.10*

1998.10.13

Freddy Strømmen

Freddy Strømmen
Seksjonsleder

Mette E. Hansen

Mette E. Hansen

PATENTSTYRET
Styret for det industrielle rettsvern

BL. - 130 - 5000 - 8.98 - Lørenskog Trykk as

PATENTSTYRET
10.NOV97 975152

ld

10. november 1997
JGS/kr
o:122336

SØKER:
Telefonaktiebolaget LM ERICSSON
S-126 25 Stockholm
Sverige

OPPFINNER:
Knut Snorre Bach Corneliussen
Hegdehaugsveien 35
0352 Oslo

TITTEL:
Fremgangsmåte for styring av trafikken i et ATM (Asynchronous Transfer Mode) nett

FULLMEKTIG:
Oslo Patentkontor, Postboks 7007M, 0306 Oslo

MEDHOLD FOR CONTROLLING THE TRAFFIC IN AN ATM (ASYNCHRONOUS TRANSFER MODE) NETWORK

Field of the invention

The present invention relates to a method for controlling the traffic in an ATM (Asynchronous Transfer Mode) network so as to maintain the Quality of Service (QoS) thereof by implementing Usage Parameter Control (UPC) comprising at least one leaky bucket unit arranged between an original cell flow of ATM-cells and a switch unit, there being used one counter for each bucket per connection, said counters being incremented and decremented according to predetermine criteria by means of timer counter means.

It is to be understood that the present invention finds particular application in connection with billing and policing in ATM based networks.

Technical background

THE PROBLEM

A widely used method for allocating resources in an ATM network is to base the allocation on the PCR (Peak Cell Rate) and the SCR (Sustainable Cell Rate). The values for PCR and SCR are provided by the user of the ATM network during the connection establishment. The values given for PCR and SCR are part of the traffic contract for the

given connection. To maintain the QoS on the user's and all the other ATM connections in the network, it is important that the traffic from the users does not exceed their PCR and SCR. The action taken to ensure that the traffic from the users is conform with the traffic contract is called the Usage Parameter Control (UPC). A method for implementing UPC is with a leaky bucket. The idea behind a leaky bucket is shown in ATM Forum's ''User-Network Interface Specification'' [1]. For Constant Bit Rate (CBR) traffic the UPC can consist of a single leaky bucket.

Figure 1 illustrates a single Leaky Bucket arrangement. The bucket is filled according to the bit rate of the traffic sent by the user. It is emptied at fixed time intervals. The size of the bucket is dependent on i.e. the PCR and CDV (Cell Delay Variation).

The leaky bucket is used to check if the user's traffic is compliant to its PCR, including the possibility of cell delay variation within an agreed bound. For Variable Bit Rate (VBR) it is proposed that the UPC consists of a dual leaky bucket. The task for the dual leaky bucket is to check that the traffic sent by the user is conform to the combination of PCR, CDV and SCR, BT (Burst Tolerance (BT) is the maximum burst size that can be sent at the SCR).

A dual leaky bucket is implemented with two buckets, one for checking PCR and CDV, and one for SCR and BT. When overflow occurs in one of the buckets, the traffic from the user is considered non conforming to the traffic contract. According to the specific network implementation the appropriate action is taken.

Figure 2 illustrates an arrangement wherein the leaky bucket (single or dual) is placed in front of the switching unit.

The problem with both the single and the dual leaky bucket is to implement them in real time systems. When the number of connections is large and a high bandwidth is used, there may be difficulties in having time to perform the various calculations (i.e. compute new bucket values). This is especially a problem when implementing a dual leaky bucket, which requires even more computations.

Known solutions

One method for implementing a dual leaky bucket is to have two buckets in parallel. There is one counter for each bucket per connection. These bucket counters are incremented every time a cell for that connections arrives, and it is checked whether the bucket counters are larger than some predefined threshold values. If one of the counter values is above its threshold, the cell is either tagged, or thrown. At regular time intervals, each bucket counter for all the connections is decremented according to a decrement value specific for each channel and bucket.

Another method for implementing a dual leaky bucket is to have two bucket counters for each connection. This method uses the same mechanism for incrementing the buckets as described above. The difference is that with this method the bucket counters for connections are not decremented at regular time intervals, only when a cell for that connection is received. To obtain a true value in each of the buckets, a time counter is used for each connection. The time counters holds the last time the bucket counters for their connection were updated.

Problems with known solutions

The problem with the first method is that the process of decrementing all the bucket counters at regular time intervals is time consuming. When the number of connections is large, high bandwidth is supported, and the time between each decrement is small, it may be impossible to have time for all these calculations.

In the second method the number of calculations is largely decreased. One problem by using this method is that you need an extra counter for each connection (the time counter). This can be a problem when the number of supported connection is high. The biggest problem with this method is the size of the time counter. When high bandwidths are supported the time counters have to be very accurate. The problem arises when a connection with much lower bandwidth than the maximum allowed bandwidth is policed. Because of the low bandwidth, cells for these connections arrive at a much higher interval than cells belonging to connections of much higher bandwidth. If the time counter is not large enough, overflow in the time counter can occur. This can lead to that cells that are conform with the traffic contract are discarded because an overflow in the time counter has occurred.

US 5 524 006 (Hluchyj et al.) relates to a second-order leaky bucket device and method for traffic management in cell relay networks, wherein the second-order leaky bucket system is utilized in connection with a peak cell rate (PCR) leaky bucket, for thereby substantially providing a predetermind quality of service.

EP-0 658 999-A2 (Dighe/NEC corporation) relates to an ATM network wherein the data frames of the system are controlled by use of ''Dual Leaky Bucket'' principle.

US 5 295 135 (Kammerl) relates to an arrangement for monitoring the bit rate in ATM networks, wherein the bit rate is monitored and controlled by means of ''Dual Leaky Bucket'' principle.

US 5 289 462 (Ahmadi et al.) relates to traffic management in packet communications networks, wherein the parameters of a ''leaky bucket'' are calculated by using a traffic metric system.

Objects of the invention

An object of the present invention is to provide a method wherein the dual leaky bucket principle can be implemented in a more efficient manner.

Antoher object of the present invention is to provide a method wherein decrementing of bucket counters can be effected as a simple and fast process.

Yet another object of the present invention is to provide a method wherein the priority of the buckets involved are utilized in a far more expedient manner.

Still another object of the present invention is to provide a method wherein the amount of needed computations are reduced substantially.

Yet another object of the present invention is to provide a method requiring less storage capacity and only one single time counter for all connections.

Still another object of the invention is to provide a method in which the decrement factor can be chosen in a more versatile manner so as to obtain better granularity of the system involved.

Brief summary of the invention

The above objects are achieved in a method as claimed in the preamble, which according to the present invention is charachterized by the combination of the following steps:
- decrementing the bucket counters at regular intervals but only when there are no arriving cells, and
- computing real bucket values for a connection when a cell for said connection arrives.

More specifically, said combination of steps are used in connection with two buckets which are arranged in the same process but given different priority, said two buckets preferrably being arranged in series.

Consequently, by placing the two buckets into the same process the amount of needed computations will be lowered.

Further, according to the present invention there is used only a single time counter for all connections involved, renedering the system even more favourable as regards computation time and accuracy.

Still further, by giving the different buckets different priority, still more time will be available for decrementing said buckets since the wasting of cells at a first bucket will allow more time for the system for decrementing the buckets involved.

Further features and advantages of the present invention will appear from the following description taken in connection with the appended drawings, as well as from the enclosed patent claims.

Brief disclosure of the drawings

Fig. 1 is a simplified diagramme illustrating the principle of a single leaky bucket arrangement, the bucket here being filled according to the bit rate of the traffic sent by the user.

Fig. 2 is a schematical diagramme illustrating an arrangement of a prior art leaky bucket principle, it being single or dual, and being placed in the front of an associated switching unit.

Fig. 3 is a schematical digramme illustrating a prior art implementation of a dual leak bucket arrangement.

Fig. 4 is a schematical diagramme illustrating an embodiment of a method according to the present invention, wherein the dual bucket principle has been implemented in the process for lowering the amount of needed computations.

Fig. 5 is a schematical block diagramme illustrating an embodiment for implementing the invention, said figure comprising the main elements included in a dual leaky bucket unit substantially as illustrated in Fig. 2.

Fig. 6 is a flow sheet illustrating the various steps taken according to the present method in order to increment for example SCR and PCR buckets.

Fig. 7 is a flow diagramme illustrating the steps involed according to the present method in order to decrement a PCR and SCR bucket involved therein.

Detailed description of embodiments

It is to be understood that the present method as been developed in connection with principally a dual leaky bucket arrangement, but it is to be understood that the principle of the present invention can also be applicable to any number of buckets operating in accordance therewith.

As mentioned previously, Fig. 1 illustrates a single leaky bucket arrangement according to the prior art. The bucket is filled according to the bit rate of the traffic sent by the user, and it is, according to prior art, emptied at fixed time intervals. The size of the bucket is dependent on i.e. the Peak Cell Rate (PCR) and the Cell Delay Variation (CDV).

In Fig. 2 there is illustrated a leaky bucket arrangement including single or dual buckets, said buckets being placed in front of the associated switching unit.

In Fig. 3 there is illustrated an example of how a prior art arrangement can be implemented, i.e. how a new cell is arrived firstly at the PCR Peak Cell Rate bucket for being checked whether compliant with the filling degree thereof, and thereafter the same new cell is controlled by the SCR Sustainable Cell Rate bucket for being checked to be compliant with also the filling degree thereof, whereafter any non-compliant signal from both buckets are sent to a decision circuit for making the decision to drop a cell and allow for a new cell to be controlled, or for the passing of said double controlled cell to be transmitted via said switching unit.

The arrangement according to Fig. 3 illustrates two buckets in parallel requiring one counter for each bucket per connection, and the associated bucket counters are incremented every time a cell for that connection arrives, and it is also checked whether the bucket counters are larger than some predefined threshold values.

According to this prior art arrangement each bucket counter for all the connections is decremented according to a decrement value specific for each channel and bucket.

As mentioned previously, another prior art method for impelemting such a dual leaky bucket is to have two bucket counters for each connection, but with this method the bucket counters for connections are not decremented at regular time intervals, only when a cell for that connection is received. To obtain a true value in each of the buckets a time counter must be used for each connection, said time counters holding the last time the bucket counters for the associated connection were updated.

Now, turning to Fig. 4, there is illustrated an embodiment of a method according to the present invention which involes a series of advantages compared with the above described prior art.

In other words, the present invention is a solution for implementing a dual leaky bucket efficiently. This invention follows some of the principles from [2], but it extends this method to support not only one, but two leaky buckets (called a dual leaky bucket). The idea is:

- Decrement the bucket counters at regular intervals (but only when there are no arriving cells).

• Compute real bucket values for a connection, when a cell for that specific connection arrives.

• Place the two buckets into the same process to lower the amount of needed computations.

• When using two or more buckets the buckets are arranged in series according to priority.

With reference to the enclosed Figures 4-7 and the enclosed appendix A there will be now given a detailed description of an example of an embodiment according to the present invention.

Firstly, reference is made to Fig. 4 illustrating a simplified basic diagramme of an embodiment according to the present invention, whereas Fig. 5 illustrates schematically an embodiment of a dual leaky bucket unit, substantially as illustrated in Fig. 2, but rearranged according to the method of the invention.

The parameters used in the following figures.

• M – The maximum number of different connections.

• m – Time counter, incremented each cell interval modulo M.

• n – The connection number.

• D – Decrement factor. This is the same for all the buckets and connections. The chosen value for D gives you the granularity of the system.

- I^{PCR}_n – Increment factor of the PCR bucket for connection n. I^{PCR}_n = bandwidth * (D/PCR).

- F^{PCR}_n – The real value of the PCR bucket for connection n. F^{PCR}_n is calculated every time a cell belonging to connection n is received.

- L^{PCR}_n – The virtual value of the PCR bucket for connection n. L^{PCR}_n is incremented by I^{PCR}_n when a cell for connection n is accepted. It is decremented by D*M every M'th cell.

- T^{PCR}_n – The threshold value of the PCR bucket for connection n . T^{PCR}_n = requested bandwidth * CDV

- I^{SCR}_n – Increment factor of the SCR bucket for connection n. I^{SCR}_n = bandwidth * (D/SCR).

- F^{SCR}_n – The real value of the PCR bucket for connection n. F^{SCR}_n is calculated every time a cell belonging to connection n is received.

- L^{SCR}_n – The virtual value of the PCR bucket for connection n. L^{SCR}_n is incremented by I^{SCR}_n when a cell for connection n is accepted. It is decremented by D*M every M'th cell.

- T^{SCR}_n – The threshold value of the PCR bucket for connection n. T^{SCR}_n = requested bandwidth * BT.

Description of Figures 4 and 5

Firstly, a cell is read from the Buffer-IN to the One cell buffer (marked ❶ in figure 5). The One cell buffer gets the VPI and VCI from the cell, and finds its connection number in the connection table. The One cell buffer then inserts the right connection number in n (marked ❷ in figure 5). The Logical Dual Leaky bucket Unit then reads the connection number from n. Then it reads the counter values related to connection n from the Counter Table (marked ❸ in figure 5). The Logical Dual Leaky Bucket Unit then calculates if the cell is compliant with the traffic contract (marked ❹ in figure 5). When the calculation is finished, the Logical Dual Leaky Bucket Unit sends the new computed counter values to Counter Table (marked ❺ in figure 5). If the cell is compliant, the Logical Dual Leaky Bucket sends a Send Cell signal to the One cell buffer (marked ❻ in figure 5). If the cell is not compliant, the Logical Dual Leaky Bucket sends a Not Send Cell signal to the One cell buffer. If the One cell buffer received a Send Cell signal from the Logical Dual Leaky Bucket, it passes the cell to the Buffer-OUT (marked ❼ in figure 5). It then reads a new cell from the Buffer-IN. If the One cell buffer received a Not Send Cell signal from the Logical Dual Leaky Bucket Unit, it reads a new cell from the Buffer-IN that overwrites the old cell.

In the enclosed Figures 6 and 7, the algorithm used to compute whether a cell is compliant to the traffic contract or not is shown. This algorithm is placed inside the Logical Dual Leaky Bucket Unit in Figure 4.

The new steps (those exceeding [2]) for supporting a dual leaky bucket will be shown in bold.

It is to be understood that Fig. 6 illustrates the steps necessary to be taken according to the invention in order to increment the SCR and PCR buckets involved in the present embodiment.

Fig. 7 illustrates the steps necessary to be taken in the illustrated embodiment in order to decrement the associated PCR and SCR bucket.

Figure 6 shows in a flow diagramme the method for incrementing the PCR and SCR bucket. After a specific time interval the process checks if a cell is waiting to be processed. If there is no cell waiting, the process goes to the decrement bucket state (see figure 7). If a new cell has arrived, the real value for the PCR bucket is calculated. This value is placed in F^{PCR}. The process then checks whether the real value (located in F^{PCR}) is greater than the maximum allowed PCR bucket value, T^{PCR}. If the real PCR bucket value is greater than the threshold value, a Not Send Cell signal is sent to the One cell buffer (see figure 6). The process then goes to state Decrement bucket (see figure 7). If the real PCR bucket value is equal or lower than the threshold value, the virtual value of the PCR bucket, L^{PCR} , is incremented by I^{PCR}. After the process has incremented the virtual value of the PCR bucket, it calculates the real value of the SCR bucket. This value is placed in F^{SCR}. It then checks whether F^{SCR} is greater than T^{SCR}. If the real value is greater than the threshold value, a Not Send Cell signal is sent to the One Cell buffer (see figure 5). If the real value of the SCR bucket is equal or lower than its threshold value, the virtual value of the SCR bucket, L^{SCR}, is calculated. A Send Cell signal is sent to the One cell buffer (see figure 5), and the process goes to the Decrement bucket state (see figure 7).

In Figure 7 the method for decrementing the buckets is shown. The first thing the process does is to increment

the time counter m. The process then calculates the virtual value of the PCR and SCR bucket for connection number m. After this calculation the process goes to the Idle state.

A pseudo code example of an implementation of the method is shown in the enclosed Appendix A. This code is written with emphasis on clarity. It is possible to run the calculation of a single bucket twice to decrease the program size

ADVANTAGES

With this invention, the number of computations is decreased, because not all buckets are decreased at regular time intervals. This method also resolves the time counter size problem, because buckets counters are decreased even though no cell has arrived on their connection. This method also requires less storage capacity because it only uses a single time counter for all the connections. This method for implementing a dual leaky bucket combines the two buckets in one process, it therefor lowers the amount of computations and overhead even more.

BROADTENDING

This method for implementing a dual leaky bucket can also be used as a single leaky bucket. You only have to set the increment value of the second bucket to zero.

REFERENCES

ATM Forum ''User-Network Interface (UNI) Specification ver. 3.1.'' af-unit-0010.002, 09/94.

U.S: Pat.No. 5 361 252 Sällberg and Larsson ''Method and device for monitoring channel split data packet transmission''

APPENDIX A

Pseudo code for the efficient dual leaky bucket implementation.

```
Begin
  Repeat
    Wait(t)
      If (New cell)
    Begin
PCR: If (m >= n) Then
        FPCRn := LPCRn - D * (m - n)
       Else
        FPCRn := LPCRn - D * (M + m -n)
       If  (FPCRn >= 0) Then
         If ((TPCRn - FPCRn) >= 0) Then
           Begin
            LPCRn := LPCRn + IPCRn          /*Cell conforming to
                                              Traffic contract*/
            End
           Else
                                            /*Cell not conforming to
                                             traffic ccntract*/
        Else
         Begin
            If (m >= n) Then
           LPCRn := IPCRn + D * (m - n)
            Else
           LPCRn := IPCRn + D * (M + m - n) /*Cell conforming to
                                              Traffic contract*/
          End
        End
SCR: If (m >= n) Then
       FSCRn := LSCRn - D * (m - n)
      Else
       FSCRn := LSCRn - D * (M + m -n)
      If  (FSCRn >= 0) Then
        If ((TSCRn - FSCRn) >= 0) Then
         Begin
           LSCRn := LSCRn + ISCRn           /*Cell conforming to
                                              Traffic contract*/
            End
           Else
            Goto DEC                    /*Cell not conforming to
                                              traffic contract*/
        Else
          Begin
            If (m >= n) Then
           LSCRn := ISCRn + D * (m - n)
            Else
            LSCRn := ISCRn + D * (M + m - n) /*Cell conforming to
                                                Traffic contract*/
          End
```

```
DEC: Begin
       m := (m + 1) MOD M
       LPCRm := LPCRm - M * D
       If (LPCRm < 0) Then
        LPCR := 0
        LSCRm := LSCRm - M * D
        If (LSCRm < 0) Then
         LSCR := 0
         End
Forever
End
```

Patent claims

1. Method for controlling the traffic in an ATM (Asyncrounous Transfer Mode) network so as to maintain the Quality of Service (QoS) thereof by implementing Usage Parameter Control (UPC) comprising at least one leaky bucket unit arranged between an original cell flow of ATM-cells and a switch unit, there being used one counter for each bucket per connection, said counters being incremented and decremented according to predetermind criteria by means of timer counter means,
characterized by the combination of the following steps:
- decrementing the bucket counters at regular intervals but only when there are no arriving cells, and
- computing real bucket values for a connection when a cell for said connection arrives.

2. Method as claimed in claim 1,
characterized in that said combination of steps are used in connection with two buckets which are arranged in the same process but given different priority, said two buckets preferrably being arranged in series.

3. Method as claimed in claim 1 or 2,
characterized in that there is used a PCR (Peak Cell Rate) bucket as a first bucket and a SCR (Sustainable Cell Rate) bucket as a second bucket, preferably connected in series with said first bucket.

4. Method as claimed in any of the claims 1-3,
characterized in that there is used a dual leaky bucket arrangement comprising an LDLBU (Logical Dual Leaky Bucket Unit) which is adapted for calculating whether an arriving ATM-cell is compliant with

the traffic contract, and which performs said calculation after having read the connection number (n) of the ATM-cell in question (cell I+0) and thereafter the counter values related to that connection (n) from a CT (Counter Table).

5. Method as claimed in claim 4, characterized in that when said calculation is finished the LDLBU will send the new computed counter values to said CT, and depending on whether the ATM-cell is compliant or not will send a Send Cell signal or Not Send Cell Signal, respectively, to a One Cell buffer being part of said dual leaky bucket arrangement.

6. Method as claimed in claim 5, characterized in that if the One Cell buffer receives a Send Cell signal from said logical dual leaky bucket it will pass the cell to a buffer-out unit, whereafter a new cell from a buffer-in unit can be read.

7. Method as claimed in claim 5, characterized in that if the One Cell buffer receives a Not Send Cell Signal from the Logical Dual Leaky Bucket Unit then it will read a new cell from said buffer-in unit that overwrites the old cell.

8. Method as claimed in any of the preceding claims, characterized in that the incrementing of the PCR and the SCR of each connection is checked at a specific time interval (m), said checking including whether there is an ATM-cell waiting to be processed, and that if no cell is waiting the bucket state will be decremented.

9. Method as claimed in any of the preceding claims, characterized in that if a new ATM-cell has arrived, then the real value of the PCR (Peak Cell

Rate) bucket is calculated, whereafter said real value is placed in the associated CT (Counter Table), the process thereafter checking whether the real value thereof is greater than the maximum allowed PCR bucket value (T^{PCR}).

10. Method as claimed in claim 9, characterized in that if the real PCR bucket value is greater than a threshold value then a Not Send Cell signal is sent to said One Cell buffer which initiates the process to go to decrement bucket state.

11. Method as claimed in claim 9 or 10, characterized in that if the real PCR bucket value is equal or lower than said threshold value then the virtual value of said PCR bucket (L^{PCR}) will be incremented by an appropriate increment factor (I^{PCR}), whereafter the process will calculate the real value of said SCR bucket which value is placed in the associated CT (Counter Table) as a real value (F^{SCR}) for said connection.

12. Method as claimed in any of the claims 9-11, characterized in that the real value (F^{SCR}) of the PCR bucket for a specific connection is checked against the value of the threshold value (T^{SCR}) of said PCR bucket for said connection, and if said real value is greater than said threshold value there will be sent a Not Send Cell signal to said One Cell buffer.

13. Method as claimed in claim 12, characterized in that if the real value of said SCR bucket is equal or lower than its threshold value, then the virtual value (L^{SCR}) of said SCR bucket is calculated and a Send Cell signal is sent to said One Cell buffer, whereafter the process goes to the decrement bucket state.

14. Method as claimed in any of the preceding claims, c h a r a c t e r i z e d i n that the decrementing of said buckets takes place by firstly incrementing said time counter (m) for thereafter calculating the virtual value of said PCR and SCR bucket, respectively, for said actual connection number (m), after which calculation the process goes to an idle state.

15. Method as claimed in claim 14, c h a r a c t e r i z e d i n that the virtual value of any PCR bucket for any connection (n) is decremented by D*M every M'th cell.

16. Method as claimed in any of the preceding claims, c h a r a c t e r i z e d i n that there is used only a single time counter for all the connections involved.

17. Method as claimed in any of the preceding claims, c h a r a c t e r i z e d i n that the increment value of a second bucket is varied according to appropriate criteria, and more specifically by setting the increment value to zero, possibly for using said method as a single leaky bucket.

A b s t r a c t

The present invention relates to a method for controlling the traffic in an ATM (Asyncrounous Transfer Mode) network so as to maintain the Quality of Service (QoS) thereof by implementing Usage Parameter Control (UPC) comprising at least one leaky bucket unit arranged between an original cell flow of ATM-cells and a switch unit, there being used one counter for each bucket per connection, said counters being incremented and decremented according to predetermind criteria by means of timer counter means, and for the objective of providing a method which can be implemented in a far more time consuming and hardware requiring implementation it is according to the invention suggested a method characherized by the combination of the following steps:

- decrementing the bucket counters at regular intervals but only when there are no arriving cells, and
- computing real bucket values for a connection when a cell for said connection arrives.

Fig. 4

1/5

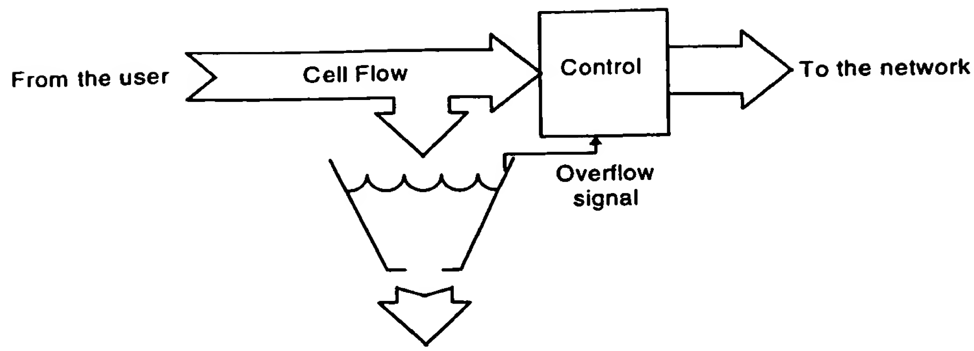

Figure 1 **A single Leaky Bucket. The bucket is filled according to the bit rate of the traffic sent by the user. It is emptied at fixed time intervals. The size of the bucket is dependent on i.e. the PCR and CDV.**

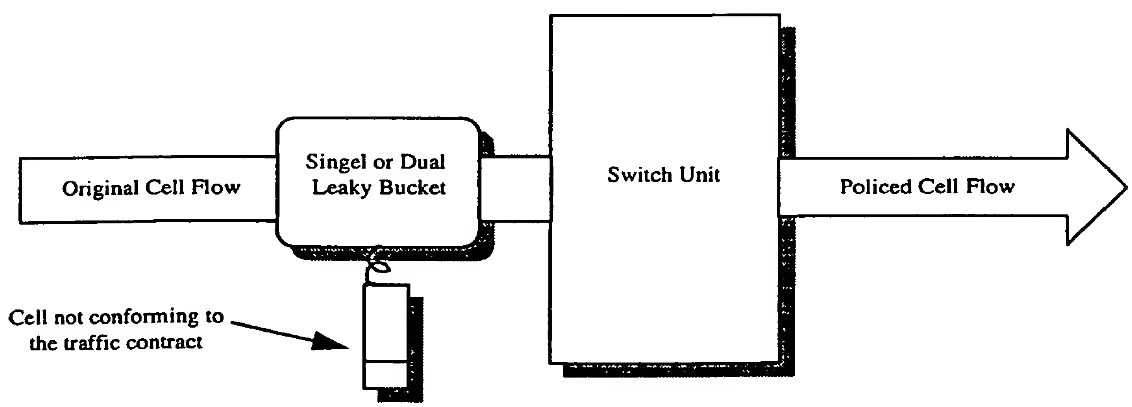

Figure 2 **The leaky bucket (single or dual) is placed in front of the switching unit.**

2/5

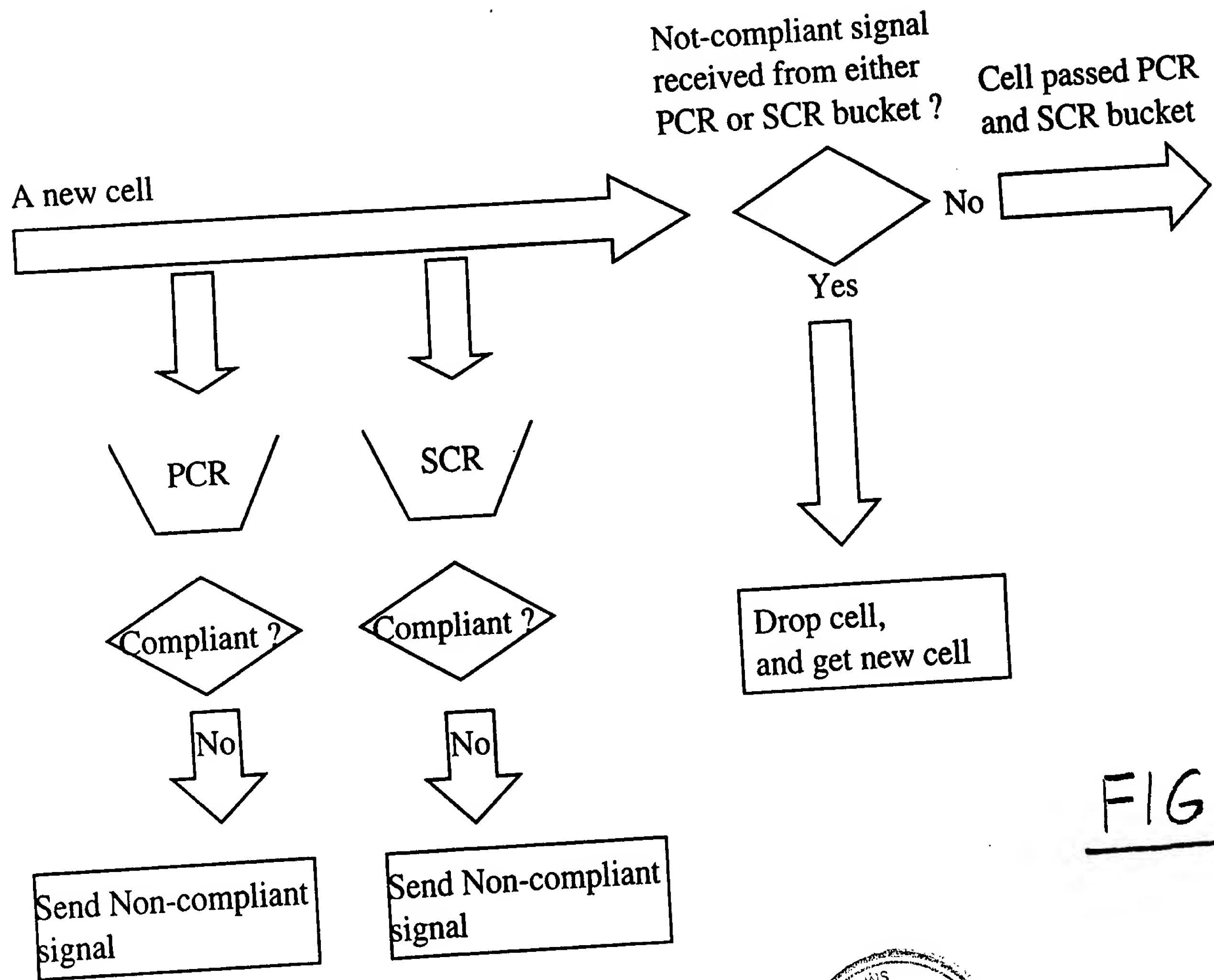
A new cell
Not-compliant signal received from either PCR or SCR bucket ?
No
Cell passed PCR and SCR bucket
Yes
Drop cell, and get new cell
PCR
SCR
Compliant ?
Compliant ?
No
No
Send Non-compliant signal
Send Non-compliant signal

FIG.3

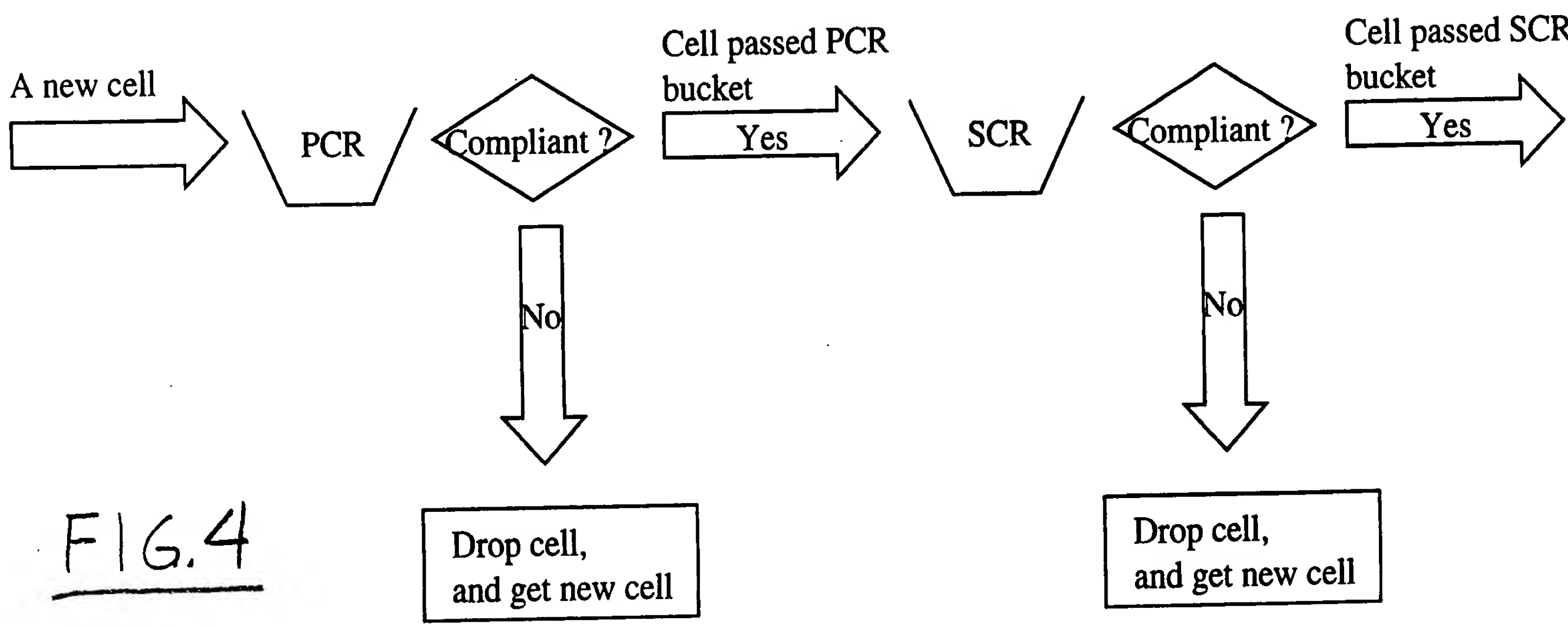
A new cell
PCR
Compliant ?
Cell passed PCR bucket
Yes
No
Drop cell, and get new cell
SCR
Compliant ?
Cell passed SCR bucket
Yes
No
Drop cell, and get new cell

FIG.4

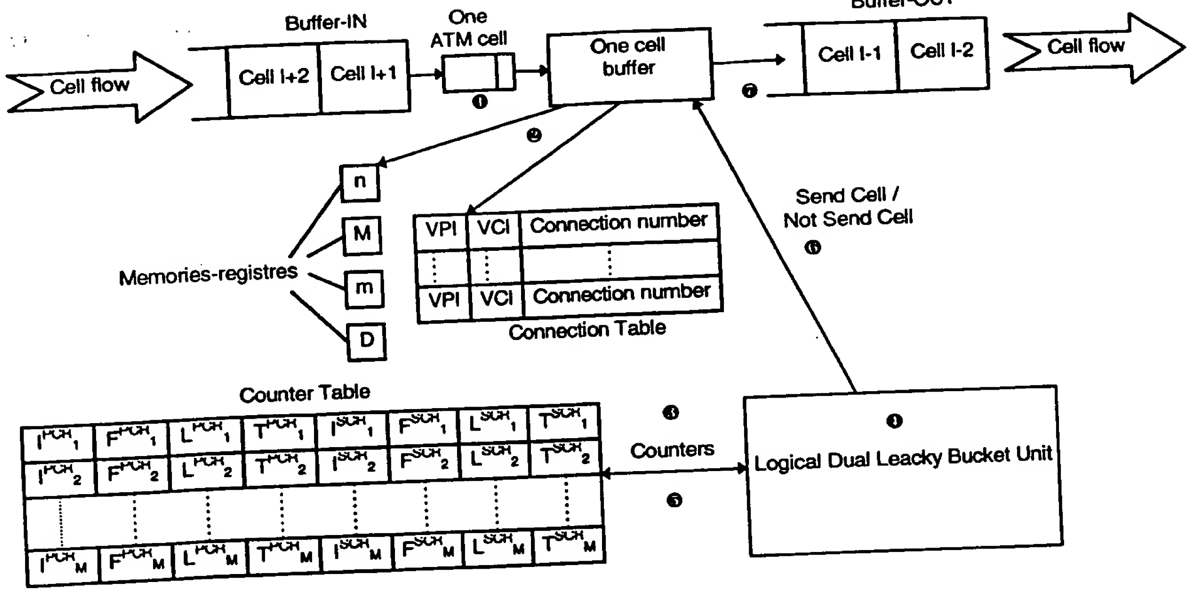

Figure ~~3~~ 5 A Schematically shown device for carrying out the invention. This figure is the inside of a Dual Leaky Bucket Unit shown in figure 2.

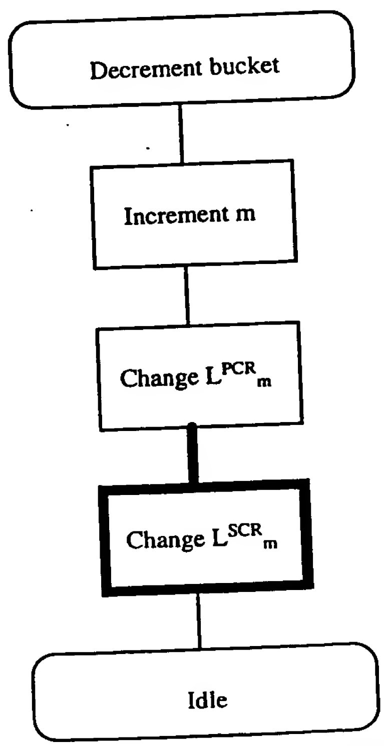

Figure ~~5~~ 7 State diagram showing the actions taken in this invention to decrement the PCR and SCR bucket.

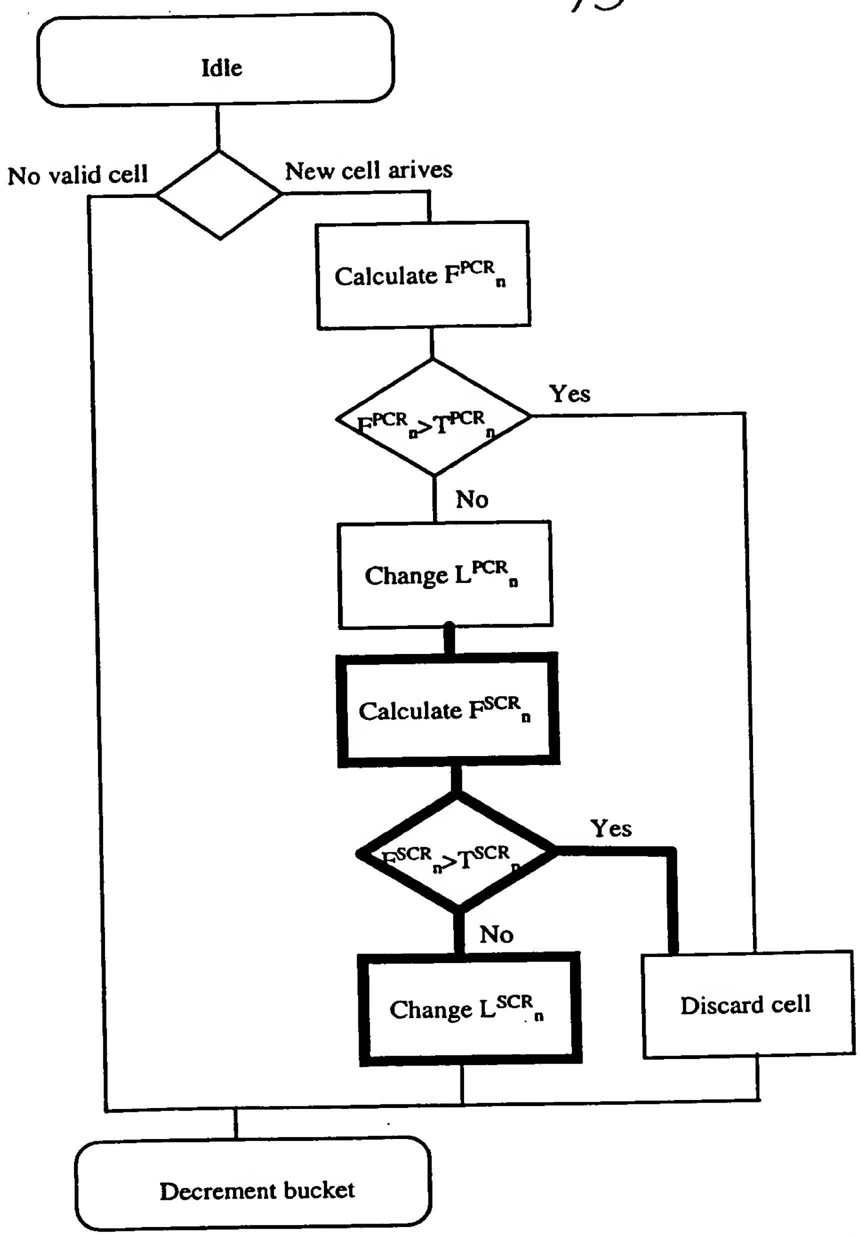

Figure ~~4~~ 6 **State diagram showing the actions taken in this invention to increment the SCR and PCR buckets.**